This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

Towards QoI and Energy-Efficiency in Participatory Crowdsourcing

Chi Harold Liu, *Member, IEEE*, Jun Fan, Pan Hui, *Member, IEEE*, Jie Wu, *Fellow, IEEE* and Kin K. Leung, *Fellow, IEEE*

***Abstract*—Today's smartphones are also fundamentally transforming the traditional understanding of "crowdsourcing" to an emerging type of participatory, task-oriented applications. It aims to support the so-called "Citizen Science efforts" for knowledge discovery, to understand the human behavior and measure/evaluate their opinions. To facilitate the above scenarios, in this paper, we propose a novel efficient network management framework for participatory crowdsourcing. Specifically, we first formulate the optimization problem and propose a closed-form, optimal solution to meet the quality-of-information (QoI) requirements of the task, while minimizing the energy consumption variance among participants. We then largely extend the traditional framework of Gur Game for distributed decision-making to recommend different levels of information contribution for each participant, by merging multiple automaton chains into a single chain with multiple steady states. By modeling the user bidding behaviors, we propose a few incentive-based participant selection schemes to maximize the platform's benefits and meet participants' expectations. We extensively evaluate the proposed schemes under the MIT Social Evolution data set, where both QoI requirements of the request and credit saving are successfully achieved, with a satisfactory level of energy consumption fairness among participants.**

***Index Terms*—Participatory crowdsourcing, Quality-of-information, Energy efficiency, Gur Game**

I. Introduction

Recent years have been witnessing the emergence of affordable, wireless and easily programmable mobile devices such as smartphones and tablets, with embedded sensors like accelerometer, gyroscope, GPS, camera and microphones [1]. These integrated rich media and location tracking features are enabling a variety of new applications and bringing forth the "participatory sensing" model [2], [3], [4] ever possible. It tasks the deployed smart devices to form interactive and participatory sensor networks to enable public and professional users to gather, analyze and share local knowledge. It is fundamentally transforming the traditional understanding of "crowdsourcing" [5], [6] to the sensory data collection in a participatory, task-oriented way. Notable examples are to support the "Citizen Science efforts" for knowledge discovery [7], a mechanism for scientific community to gather data from the public in a distributed way to understand the human behavior and measure/evaluate their opinions.

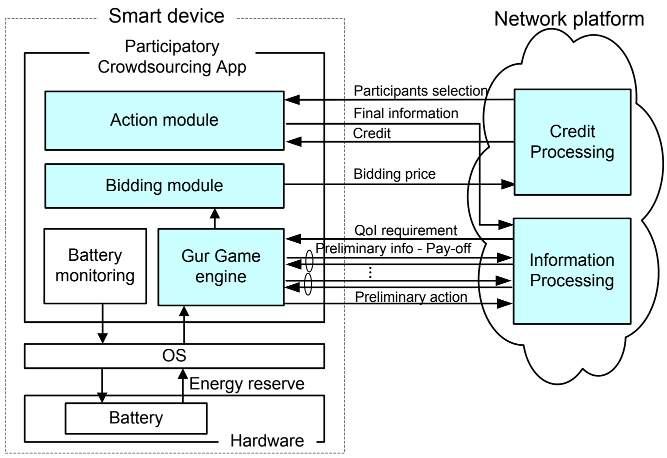

Fig. 1. The considered participatory crowdsourcing system, where a network platform recruits smartphone users to provide sensing services and pays them incentives as the reward for future participation. The light blue boxes and the algorithms are proposed in this paper.

Traditional methods like self-reported surveys and experience sampling often suffer from the subjectivity and memory effects, and the frustration associated with real-time, laborious discussion of multiple opinions, lack of consensus, and the feeling of "wasting time" [8]. In our case, the required information can be achieved via a participatory crowdsourcing platform on the smart device to provide quick and instantaneous information gathering among the collocated group of people, presenting a tempting alternative. As shown in Fig. 1, the participatory crowdsourcing system consists of a sensing network platform, which includes an information center and a credit center and resides in the cloud, and many registered smartphone users, who are connected with the platform via the existing cellular network infrastructure and provide sensing services to the platform. The platform propagates questionnaire request to the recruited smartphone users. Upon receiving the query, they decide whether and how to respond distributedly, and send the replies back to the platform, where data processing may occur before a result is finally obtained. The users who supplied data would receive some form of credit from the platform as a reward for supporting the efficacy of the application. This is essentially

This work is financially sponsored by National Natural Science Foundation of China (Grant No. 61300179).

C. H. Liu and J. Fan are with School of Software at Beijing Institute of Technology, China. C. H. Liu is the Corresponding Author of this paper. E-mail: {chiliu,jfan}@bit.edu.cn

P. Hui is with Hong Kong University of Science and Technology and Deutsche Telekom Laboratories, Germany. Email: panhui@cse.ust.hk

J. Wu is with Temple University, PA, USA. Email: jiewu@temple.edu

K. K. Leung is with EEE and Computing Departments at Imperial College, London, UK. Email: kin.leung@imperial.ac.uk

Early results were reported at IEEE SECON'11, Salt Lake City, USA.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

different from the online social networks, where users may not be associated with similar background information, and the questionnaire is not specifically designed for a small, co-located community to understand a particular aspect like their health condition.

Supporting applications such as the one above requires addressing the following challenge: *how to manage the participants distributedly to achieve the quality-of-information (QoI) required by the questionnaire request, while providing satisfactory benefits to the network platform and participants*? QoI relates to the ability to judge whether information is *fit-for-use* for a particular purpose [9], [10]. For the purpose of this paper, we assume that QoI is characterized by a set of attributes that quantize the amount of information required by the request. Since smart devices are not dedicated sensing devices and have essential demands for energy resources, such as voice calls, how to balance the information contribution with their energy reserve is an open issue, and more importantly, to design a distributed algorithm without centralized control to meet the mobility requirement of participants. In addition, users participating in such crowdsourcing environment also expose themselves to potential privacy threats. Therefore, users might not be willing to use their resources and participate unless they receive something in return. This brings forth the important issue of incentive-based techniques, acting as the driving force to motivate user participation, provide sufficient and continuous influx of user contributions and guarantee good QoI. These challenges and general approach serve as the basis for our work.

Building upon and significantly extending our previous work [11], we propose a novel efficient network management framework for participatory crowdsourcing. The contribution of this paper is five-fold: First, considering the QoI requirements of the questionnaire request and the residual energy state of all participants, we formulate an optimization problem to achieve the highest degree of energy consumption fairness, and subsequently propose a closed-form solution with low computational comlexity , which has not been considered in [11]. Second, we propose a distributed, QoI-aware participatory crowdsourcing framework embedded in each smart device that recommends the amount of information contribution in aware of its energy consumption. Here, we extensively extend [11] by merging multiple automaton chains into one single chain with multiple steady states, to represent different amount of contributed information. To meet the requirements of both QoI and energy fairness, we propose a two-step decision making process. Third, we explicitly design an incentive-based participant selection mechanism to motivate users' participation, while maximizing the benefit of the network platform by minimizing the total paid credits to participants. In [11], the goal is to minimize the necessary adaptation of the pricing scheme. Then, we provide thorough analysis on the proposed user selection and credit allocation method, given different presumed user bidding behaviors. Finally, we conduct extensive simulations on a real social data set, largely enriching and deepening the exploration of framework performance. Extensive results confirm that the proposed scheme successfully achieves the QoI requirement while providing a satisfactory level of energy consumption fairness among participants, with negligible computational complexity. On the other hand, the total paid credits are effectively reduced with the proposed auction-based approach, which guarantee a certain level of benefit for the network platform.

TABLE I
SUMMARY OF IMPORTANT SYMBOLS

Symbol	Definition (Section where the symbol is first used)
$\mathcal{N}$	the set of participants of size N (III-A)
L	the number of QoI attributes (III-A)
$u^r(l)$	required value of the l-th QoI attribute (III-A)
$u_{\min}(l)$	lower bound for $u^r(l)$ (III-A)
$u_{\max}(l)$	upper bound for $u^r(l)$ (III-A)
$u_i^a(l)$	the amount of information contribution by user i for the l-th QoI attribute (III-A)
E_i	initial energy reserve of user i (III-A)
$\bar{E}_i$	remaining energy of user i upon receiving request (III-A)
ξ_i	total energy consumption ratio of user i (III-A)
γ	normalization factor for ξ_i (III-A)
$f_l(\cdot)$	information fusion algorithm for the l-th QoI attr. (III-A)
$u^a(l)$	collective contribution of all users for l-th QoI attr. (III-A)
p_i	answering probability of user i for "average" option (IV-B)
u_i	the actual number of answers reported by user i (IV-B)
$\{u_i\}^*$	optimal number of answers users should return (V)
$\mathcal{S}$	set of steady states (V-B)
$\mathcal{T}$	set of transitional states (V-B)
σ_j	set of transitional states subordinate to steady state j (V-B)
$S_i(k)$	current state of automaton user i resides in k-th step (V-B)
β	the number of subordinate states of each steady state (V-B)
K	the maximum iteration step (V-B)
$I(k)$	QoI index at k-th iteration step (V-C)
η	scaling factor for QoI index (V-C)
$R_i(k)$	reward prob. user i receives at k-th iteration step (V-D)
$P_i(k)$	penalty prob. user i receives at k-th iteration step (V-D)
α	scaling factor for $R_i(k)$ (V-D)
ς	the range for satisfactory QoI in Q-step (V-E)
ϵ	the adjustment threshold in V-step (V-E)
φ	total paid credits to participants (VI)
δ	QoI margin for user selection (VI)
b_i	bidding price of user i (VI)
κ	adaptation ratio for bidding(VI)
χ	bid change step size (VI)

The rest of the paper is organized as follows. Section II highlights the related research activities. Section III presents a formal model of our system and shows the system flow. Section IV describes the MIT Social Evolution data set with our treatment for participatory crowdsourcing, and the system pertinent solutions based on the extended Gur Game structure is presented in Section V. Section VI presents the incentive-based participant selection mechanism. Extensive results are given in Section VII. Section VIII discussed a few practical issues related to our framework. Finally, a conclusion is drawn in Section IX.

A summary of important symbols used in this paper is listed in Table I.

II. RELATED WORK

Plenty of participatory sensing applications across different areas have been proposed. EasyTracker [12] is designed for transit tracking, mapping, and arrival time prediction by deploying smartphone GPS unit on each vehicle. The CenceMe project [13] investigates the use of phone sensors to classify events in people's lives, and selectively share the presence using online social networks such as Twitter and Facebook. The "Micro-Blog" [14] allows smartphone-equipped users to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

generate and share multimedia through social participation. In [15], "Hapori" is proposed as a context driven local search framework built on the community behaviors and user similarity modeling. [1] proposes "EmotionSense", a mobile sensing platform for social psychology studies based on mobile phones, including the ability of sensing individual emotions as well as activities, verbal and proximity interactions among members of social groups. "CoMon" is proposed in [16] as a cooperative ambience monitoring platform to monitor the environment by user cooperations. It leverages the encounter history to measure the social relationships among users and estimates the potential cooperation duration for candidate cooperators. [17] discusses the feasibility of "Citizen Mapping", by using the crowdsourced GIS data to evaluate the environmental justice and equality. [18] addresses the noise pollution monitoring based on data collected by GPS sensors and microphones in smartphones. In [6], the authors investigate a crowd-voting case study, where a web-based t-shirt company "Threadless" selects the products it sells by having users provide designs and vote on the ones they like. [8] discusses the applicability of building a distributed voting application based on mobile ad hoc networks, allowing users to efficiently express preferences in a timely manner.

Regarding the energy-aware sensor networks management, [19] is the first work to use the mathematical paradigm of the Gur Game [20], [21], [22] to dynamically adjust the optimal number of sensors to operate through a few steps of iteration. Later, it is extended in [23], where an energy-aware algorithm is developed, and the periodic sleeping mechanism is introduced. [24] uses a Gur Game formulation to maximize the number of regions covered by sensors.

As for the design of incentive mechanism for participatory sensing/crowdsourcing, [25] proposes a reverse auction-based dynamic pricing incentive mechanism with virtual credit, to minimize the total incentive cost. In [26], the authors design an incentive mechanism for the user-centric model, using auction-based approach to guarantee user participation. The authors in [27] consider the double roles of a user in participatory sensing, and propose a demand-based approach to maximize fairness and social welfare. In comparison, we aim to maximize the platform's benefits, given the presumed user behaviors. Our approach is also integrated with the framework of Gur Game for distributed decision-making, thus not only maximizing the benefits from economical perspectives, but also meeting the QoI requirements of sensing tasks.

III. System Model

In this section, we first present a formal model for describing the participatory crowdsourcing application, and then introduce the system flow.

A. Assumptions and Notations

We consider the application of opinion/preference gathering, where the local network consists of N volunteer contributors (or participants) of a set denoted by $\mathcal{N} \triangleq \{i|i = 1, 2, \ldots, N\}$. The questionnaire query is associated with a set of L QoI attributes, where $\underline{u}^r = \{u^r(l)|l = 1, 2, \ldots, L\}$ with superscript "r" denote the "required" value. Without loss of generality, we assume that every element in $\underline{u}^r$ is countable and can be quantized by a metric with upper and/or lower bound, denoted as $u_{\min}(l), u_{\max}(l), \forall l$. For instance, $u^r(1), u^r(2), u^r(3)$ can represent the required number of answers sufficiently large for statistical analysis, the required image resolution and duration for a contributed video, respectively. Then, the request is propagated through the existing network infrastructure and delivered to the participants. For example, in the cellular network settings the BS collects the feedback from all participants, working under the standardized communication protocol, like GSM/3G/LTE.

We denote $u_i^a(l) \in [u_{\min}(l), u_{\max}(l)], \forall l = 1, 2, \ldots, L$ as the amount of information contributed by user i corresponding to the l-th QoI attribute. Superscript "a" represents the "attained" value of the attribute. An example is the actual number of answers reported by the i-th participant among multiple times of questionnaire deliveries. Furthermore, we assume that all participants' smart devices have enough energy reserve to complete one request, and for each user i, its initial energy reserve of the smart device is denoted as E_i, and the remaining energy upon receiving the request is denoted as $\bar{E}_i, \forall i \in \mathcal{N}$. We further introduce a scaling factor γ to normalize the amount of information contribution, to denote the proportional amount of energy consumption due to information contribution. Then, after the task, the total percentage of energy consumption for user i is computed as:

$$\xi_i = 1 - \frac{\bar{E}_i}{E_i} + \sum_{l=1}^{L} \left(\frac{u_i^a(l)}{\gamma} \right), \tag{1}$$

where $\xi_i \in [0, 1], \ \forall i \in \mathcal{N}$. The first part of (1), $1 - \bar{E}_i/E_i$, defines the energy consumption percentage before receiving the crowdsourcing request. The second part of (1) indicates the consumed energy percentage when responding the crowdsourcing request, which is proportional to the information contribution towards the total L QoI attributes.

Finally, we use the mapping $f_l, \forall l$ to denote a set of information fusion algorithms like the one reported in [28] corresponding to the required QoI $u^r(l)$. It aggregates multiple information sources obtained from all participants to a single view of the event,

$$u^a(l) = f_l(u_i^a(l)), \quad \forall i \in \mathcal{N}, \ l = 1, 2, \ldots, L. \tag{2}$$

We call a specific QoI requirement is satisfied, if and only if, $u^a(l) \geq u^r(l), \forall l = 1, 2, \ldots, L$.

If the property is considered to be the same among all users, a straightforward example is $f_l \triangleq \sum_i$ to collect answers reported from all participants:

$$u^a(l) = \sum_{i=1}^{N} u_i^a(l). \tag{3}$$

If we consider the different properties of each user, examples of f_l can be:

$$u^a(l) = \sum_{i=1}^{N} p_i u_i^a(l). \tag{4}$$

$$u^a(l) = \max\{u_i^a(l)\}, \ \forall i \in \mathcal{N}. \tag{5}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

Particularly in (4), parameter $p_i, \forall i \in \mathcal{N}$ exactly describes the property of each user, connecting the contributed information with its recognized value. At the same time, it also indicates the QoI difference among users. On the basis of (4), if we consider the decreasing marginal returns in the amount of information, another example can be:

$$u^a(l) = \vartheta^{\sum_{i=1}^{N} p_i u_i^a(l)/\varpi} \cdot \sum_{i=1}^{N} p_i u_i^a(l), \tag{6}$$

where $\vartheta \in (0, 1]$ is the decay coefficient, and ϖ is the scaling factor.

As our proposal is based on the mathematical paradigm of Gur Game, our solution is transparent to any specific form of function used in the QoI model, whether or not it is discontinuous, multimodal, or concave, etc. In this paper, we adopt the fusion functions in (4) and (6), and verify the adaptability of our solution in the Section VII.

B. System Flow

Gur Game [20], [21] was proposed to use in distribute systems who wish a collection of agents to cooperate on a task. Each agent is associated with a finite state automaton that independently guides the agent's action, while taking into account the collective feedback that eventually captures the composite effect of all agents' actions. Compare to our considered participatory crowdsourcing scenarios, the participant's smart device in this case acts as the "agent", where the associated automaton can be easily deployed through a piece of software in the mobile OS. The "task" translates exactly to our focused social studies crowdsourced from a co-located group of participants; and the "composite effect of all agents' actions" is then the result of the participants' action upon returning answers to the querier. Therefore, we believe that the fundamentals of Gur Game serve as the ideal engine algorithmically due to its robustness, simplicity and decentralized features. To make it particularly suitable for crowdsourcing, in this paper we largely extend the existing Gur Game and use it as part of the overall system flow shown in Fig. 1.

The system flow consists of two stages. The first stage relates to the interaction between the information center of the network platform and the Gur Game engine programmed in user's smart device. The inputs are the user's smart device residual energy levels and multiple QoI requirements of the request. At each iteration step of the Gur Game, the smart device sends its preliminary action (as a result of our proposed Gur Game algorithm) back to the platform. The latter then calculates the pay-off value based on the received collective pieces of information from all participants, and propagates it back to Gur Game engine of each user. Based on this feedback, the automaton changes its current state and generates the new action, i.e., the level of information contribution. Therefore, the Gur Game engine uses the trial-and-error method to produce the best result at each step and iteratively achieves the overall optimum to fulfill the QoI requirements in aware of energy efficiency.

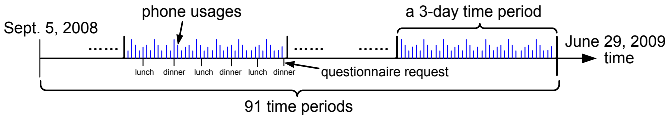

Fig. 2. The illustration of quantizing the entire duration into 91 time periods and when considered participatory crowdsourcing request arrives.

The second stage relates to the interaction between the credit center of the network platform and the bidding module of user's smart device. After the first stage, the platform obtains each user's preliminary action. Then, the users send their bids to the platform, which represent their expected paid credits for unit amount of information contribution. The network platform selects the users who can meet the QoI requirements of the request and help reduce total paid credits as the active users for questionnaire answering, receives their final information contribution determined by the Gur Game engine in the first stage, and pay their credits according to the previous bids. We next describe a detailed implementation and solution of this framework.

IV. A Case Study

This section first provides an overview of the used Social Evolution data set gathered by MIT Media Lab [29], and then illustrates how we use it to motivate our participatory crowdsoucing application. The data set is generated by an application on 80 undergraduates' smart devices, who move around the campus. It collects the phone usages and student locations from October 2008 to June 2009. The phone usage data consist of 3.15 million records of Bluetooth scans, 3.63 million scans of WLAN access-points, 61,100 call records, and 47,700 logged SMS events. Also, students provide offline, self-report answers related to their health habits, diet and exercise, weight changes, and political opinions during the presidential election campaign. In our simulation, we use the phone usage data, and the self-report answers on the health condition to motivate and form the participatory crowdsourcing process as described below.

A. Phone Usage Data

We extract the phone usage records from September 5, 2008 to June 29, 2009, in a total of 273 consecutive days. The data include 49,906 voice call records with the calling time, duration, caller and callee information, 33,148 SMS events including the sending time, sender and receiver information. Then, to facilitate our simulation, we quantize the entire 273 days into 91 independent time periods each of which involves a 3-day usages, as shown in Fig. 2. We assume that phone usages between two consecutive periods do not interrelate with each other, and since the phone battery status is not provided in the data set we simply assume that at the beginning of any time period the device is fully charged. As a result, we have 91 time periods, each of which is associated a set of phone usages, and this is used to evaluate how different phone data impact on the proposed algorithm. Besides, we also make the following two assumptions:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

- a fully charged phone battery can afford a 5-day standby time, a 6-hour talking time, or 10,000 SMS events;
- for simplicity, apart from voice call, SMS, and standby, we neglect all other situations for phone energy consumption.

Therefore, we can compute the phone's remaining energy $\bar{E}_i$, $\forall i \in \mathcal{N}$ for each time period.

B. Considered Participatory Crowdcourcing Scenario

From the data set, students report their health condition periodically, including the weight, height, the amount of salad per week and fruit per day, the number of aerobic exercises per week, the amount of smoking, and most importantly, how they evaluate their health condition among: "very unhealthy", "unhealthy", "below average", "average", "healthy", and "very healthy". In the survey, each student is requested to report 6 times in total, however in practice the received number of returned questionnaire diverse individually. From 382 total collected records, we observe that 6 answers are reported as "very unhealthy", 50 as "unhealthy", 88 as "below average", 148 as "average", 82 as "healthy", and 8 as "very healthy". Without loss of generality, we assume that the above number of collected answers are just about sufficient for further statistical analysis, and serves as the basis of our simulation.

Since these results are obtained at offline, in our simulation, we are particularly interested in *evaluating the student health condition by collecting answers for a specific item as above (e.g., "average") from the targeted participatory crowdsourcing platform*. The answers are simply one of the above options, and can be delivered by an SMS. Therefore, the amount of energy consumption in reporting the answer can be computed by (1). Then, QoI requirement is considered as *the minimum required number of answers collected from participants* which is sufficient for statistical analysis. That is, for "average" option, we have $u^r = 148$ and we drop the index $l = 1$ as only one QoI requirement is considered. Furthermore, from the reports 47 students voted this option, and they are considered as the participants of set $\mathcal{N}$. Also, the probability of each student to rate the "average" option (among other options) is different; and thus, for user $i \in \mathcal{N}$, we compute an "answering probability", denoted as $p_i, \forall i$, as a fraction of answers towards the considered "average" option. Finally, to be consistent with the actual reports, we set the upper and lower bound of the number of answers each participant is allowed to report as $u_{\text{max}} = 6$ and $u_{\text{min}} = 0$. This corresponds to the delivery of $u_{\text{max}} = 6$ questionnaires to all students, and aims to the collection of student's health condition two times a day, based on their diet habit for lunch and dinner, as shown in Fig. 2.

V. Problem Formulation and Optimal Solution

Let $u_i^a = u_i,\ \forall i \in \mathcal{N}$, where u_i denotes the actual number of answers reported by user i, i.e., an output from the Gur Game engine, and $0 \le u_i \le u_{\text{max}},\ u_i \in \mathbb{Z},\ \forall i \in \mathcal{N}$. Then, our goal is to find optimal number of answers participants are recommended to return, denoted as $\{u_i\}^*$. For the information quality model, here we use the fusion function in (4). In our case study, the "answering probability" p_i is calculated from the data set. While in actual crowdsourcing systems, it can be determined by studying the history of user contribution behaviours. Given the answering probability p_i and the total energy consumption ratio ξ_i (from both phone usages and participatory crowdsourcing), $\forall i \in \mathcal{N}$, each participant aims to return an "optimal" number of answers to total delivered $u_{\text{max}} = 6$ questionnaires that collectively form the vector $\{u_i\}^*$. By "optimal", we mean that $\{u_i\}^*$ should not only achieve the required QoI, but also offer a satisfactory level of energy consumption fairness, in terms of minimizing the variance of energy consumption ratio for all participants, denoted as V. Towards this end, we formulate the optimization problem as:

$$\begin{aligned} \underset{\{u_i\}}{\textit{minimize:}} \quad & V = \frac{1}{N}\sum_{i=1}^{N}\left(\xi_i - \frac{1}{N}\sum_{i=1}^{N}\xi_i\right)^2 \\ \textit{subject to:} \quad & \sum_{i=1}^{N} p_i u_i = u^r, \\ & 0 \le u_i \le u_{\text{max}},\ u_i \in \mathbb{Z},\ \forall i \in \mathcal{N}, \end{aligned} \tag{7}$$

The first constraint is to guarantee the received number of answers are not smaller than the required value with satisfactory QoI; and the second constraint is due to the number of delivered questionnaires is maximally u_{max}. The optimization problem in (7) is a non-linear integer programming problem and it is NP-hard.

Theorem 5.1: The closed-form solution of optimization problem in (7) can be mathematically obtained by trying combinations of three categories of the $\{u_i\}$ values.

Proof: Without loss of generality, we consider a generalized version of Theorem 5.1, where $u_i \in \mathbb{R}$. Combining (1) and (7), we have:

$$NV = \sum_{i=1}^{N}\left(1 - \frac{\bar{E}_i}{E_i} - \frac{1}{N}\sum_{i=1}^{N}\left(1 - \frac{\bar{E}_i}{E_i}\right) + \frac{p_i u_i}{\gamma} - \sum_{i=1}^{N}\frac{p_i u_i}{N\gamma}\right)^2.$$

Let $q_i = 1 - \frac{\bar{E}_i}{E_i} - \frac{1}{N}\sum_{i=1}^{N}\left(1 - \frac{\bar{E}_i}{E_i}\right)$. Then,

$$NV = \sum_{i=1}^{N}\left(q_i + \frac{p_i u_i}{\gamma} - \sum_{i=1}^{N}\frac{p_i u_i}{N\gamma}\right)^2 = \frac{N-1}{N\gamma^2}\left(\sum_{i=1}^{N} p_i u_i\right)^2 + \sum_{i=1}^{N} q_i^2 - \frac{2}{\gamma^2}\sum_{i \ne j} p_i u_i p_j u_j + \frac{2}{\gamma}\sum_{i=1}^{N} q_i p_i u_i.$$

Given u^r, in order to satisfy the constraint, we have $\sum_{i=1}^{N} p_i u_i = u^r$. Therefore, the first two items can be treated as constants. Hence, minimizing V is equivalent to:

$$\begin{aligned} \underset{\{u_i\}}{\textit{maximize:}} \quad & \frac{2}{\gamma^2}\sum_{i \ne j} p_i u_i p_j u_j - \frac{2}{\gamma}\sum_{i=1}^{N} q_i p_i u_i, \\ \textit{subject to:} \quad & \sum_{i=1}^{N} p_i u_i = u^r,\ 0 \le u_i \le u_{\text{max}},\ \forall i \in \mathcal{N}. \end{aligned} \tag{8}$$

According to the Kuhn-Tucker condition [30], the Lagrangian is:

$$F = \frac{2}{\gamma^2}\sum_{i\neq j} p_i u_i p_j u_j - \frac{2}{\gamma}\sum_{i=1}^{N} q_i p_i u_i + \lambda(u^r - \sum_{i=1}^{N} p_i u_i) + \sum_{i=1}^{N}\mu_i(u_{\max} - u_i) + \sum_{i=1}^{N}\omega_i u_i,$$

So we need:

$$\frac{\partial F}{\partial u_i} = 0, \lambda(u^r - \sum_{i=1}^{N} p_i u_i) = 0, \mu_i(u_{\max} - u_i) = 0, \omega_i u_i = 0,$$

where

$$\lambda \geq 0,\ \mu_i \geq 0,\ \omega_i \geq 0,\ 0 \leq u_i \leq u_{\max},\ \forall i \in \mathcal{N}. \tag{9}$$

Then, we break the analysis into 18 cases based on the complementarity conditions of λ, μ and ω values. Checking them one by one according to the constraints in (9), finally there remains three cases which can be classified as three categories of $\{u_i\}$ values:

- **Category 1:** $\lambda \neq 0$, $\mu_i = 0$, $\omega_i = 0 \Rightarrow u_i \in (0, u_{\max})$,
- **Category 2:** $\lambda \neq 0$, $\mu_i = 0$, $\omega_i \neq 0 \Rightarrow u_i = 0$,
- **Category 3:** $\lambda \neq 0$, $\mu_i \neq 0$, $\omega_i = 0 \Rightarrow u_i = u_{\max}$.

Then, we divide the set $\mathcal{N}$ into three subsets, $\mathcal{N}_1 = \{i \mid u_i \in (0, u_{\max}),\ i \in \mathcal{N}\}, \mathcal{N}_2 = \{i \mid u_i = 0,\ i \in \mathcal{N}\}, \mathcal{N}_3 = \{i \mid u_i = u_{\max},\ i \in \mathcal{N}\}$, where $\mathcal{N}_1 \cup \mathcal{N}_2 \cup \mathcal{N}_3 = \mathcal{N}$. Therefore, by trying total 3^N combinations of categories of $\{u_i\}$ values, we can get the desired $\{u_i\}^*$ which achieves the minimum variance of energy consumption ratio. Next, we derive the corresponding equations for trying combinations.

For a given situation of $\mathcal{N}_1$, $\mathcal{N}_2$, $\mathcal{N}_3$, we have:

$$u^r = \sum_{i\in\mathcal{N}_1} p_i u_i + \sum_{i\in\mathcal{N}_2} p_i u_i + \sum_{i\in\mathcal{N}_3} p_i u_i = \sum_{i\in\mathcal{N}_1} p_i u_i + \sum_{i\in\mathcal{N}_3} p_i u_{\max}, \tag{10}$$

From $\frac{\partial F}{\partial u_i} = 0$, we get:

$$\frac{2}{\gamma^2} p_i(u^r - p_i u_i) - \frac{2}{\gamma} q_i p_i - \lambda p_i - \mu_i + \omega_i = 0, \tag{11}$$

Category 1: For $i \in \mathcal{N}_1$, because $\mu_i = 0$, $\omega_i = 0$, from (11) we get the sum as:

$$\sum_{i\in\mathcal{N}_1}\left(\frac{2}{\gamma^2}(u^r - p_i u_i) - \frac{2}{\gamma} q_i - \lambda\right) = 0. \tag{12}$$

Then, from (10) and (12), we get:

$$\lambda = \frac{2|\mathcal{N}_1|u^r - 2u^r + 2u_{\max}\sum_{i\in\mathcal{N}_3} p_i - 2\gamma\sum_{i\in\mathcal{N}_1} q_i}{|\mathcal{N}_1|\gamma^2}. \tag{13}$$

Putting (13) back to (12), we have:

$$u_i = \frac{u^r + \gamma\sum_{i\in\mathcal{N}_1} q_i - u_{\max}\sum_{i\in\mathcal{N}_3} p_i}{|\mathcal{N}_1| p_i} - \frac{\gamma q_i}{p_i},\ \forall i \in \mathcal{N}_1. \tag{14}$$

Category 2: For $i \in \mathcal{N}_2$, because $\mu_i = 0$, $\omega_i \neq 0$, $u_i = 0$, from (11), we get:

$$\omega_i = \frac{2}{\gamma} q_i p_i + \lambda p_i - \frac{2}{\gamma^2} u^r p_i. \tag{15}$$

Category 3: For $i \in \mathcal{N}_3$, because $\mu_i \neq 0$, $\omega_i = 0$, $u_i = u_{\max}$, from (11), we get:

$$\mu_i = \frac{2}{\gamma^2} u_r p_i - \frac{2}{\gamma^2} u_{\max} p_i^2 - \frac{2}{\gamma} q_i p_i - \lambda p_i. \tag{16}$$

Then, we check if the inequalities in (9) are satisfied. If satisfied, we record the $\{u_i\}$ values and its corresponding variance V. Therefore, the optimal $\{u_i\}^*$ and minimum V are obtained by trying possible combinations of sets $\mathcal{N}_1, \mathcal{N}_2, \mathcal{N}_3$. ■

It is worth noting that the optimization problem in (7) can also be solved by the popular optimization toolbox like YAMLIP [31]. It offers a near-optimal solution with tunable absolute error margin. Comparing with it, our proposal in Theorem 5.1 is of low computational complexity, by just computing a few mathematical equations. In contrast, YAMLIP alike toolboxes consist of tens of thousand lines of codes and usually cannot be implemented in resource-constrained smart devices.

A. The Gur Game

Gur Game was first used to power on a desired number of sensors in WSNs [19], where each sensor is associated with a finite discrete-time automation. The automaton is a single nearest-neighbor Markov chain of memory size $2M$. Starting From the left-most state, the states are numbered from $-M$ to -1, then followed by 1 to M until the right-most state. The negative numbered states represent the "idle" action, while positive numbered states represent the "active" action. Each sensor makes state transition decisions distributedly according to the pre-defined pay-off structure, which consists of a pay-off function with bounded value $[0, 1]$. Based on the new states of sensors, the pay-off value is updated after each iteration. Note that the reward function reaches its peak when desired number of sensors stay at "active" decision states. The states of different sensors gradually converge after a few steps of iterations.

It is clear that the original Gur Game only offers two candidate decisions for each participating sensor: idle or active. However, in our considered participatory crowdsroucing application each user may answer the questionnaire request multiple times from 0 to $u_{\max}$. Thus, we need to extend the original Gur Game to a new form, by merging multiple automaton chains into a single chain with multiple steady states, where each state represents a participating action, i.e., the number of answers in our case.

B. Extended Gur Game Structure

Without loss of generality, let K denote the maximum number of iterations every Gur Game automaton allows to run before recommending the action, which impacts the speed of system convergence and will be evaluated in the next section. Our proposed Gur Game structure consists of two kinds of states: the steady states $\mathcal{S} = \{j|j = 0, 1, \ldots, u_{\max}\}$ and the transitional states $\mathcal{T} = \bigcup_{j=0}^{u_{\max}} \sigma_j$, where σ_j denotes a set of transitional states that are subordinate to state $j \in \mathcal{S}$. By referring to "subordinate", we mean that the action $u_i(k)$ (i.e.,

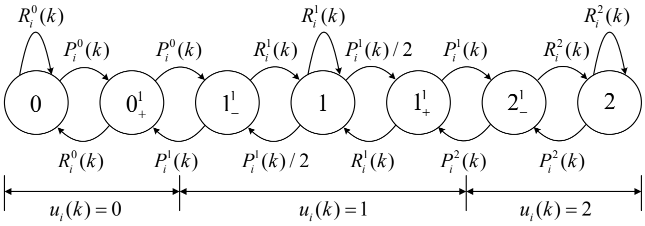

Fig. 3. An example of the extended Gur Game structure when $u_{\text{max}} = 2$, $\beta = 2$. Each steady state has at most 2 subordinate transitional states. The arrow shows the direction of state transition under the reward or penalty probability.

the number of answers to the query) behind the state j and σ_j in the k-th step of iteration are exactly the same, or:

$$u_i(k) = j, \quad \text{if } S_i(k) \in j \cup \sigma_j, \ \forall i, j, k, \tag{17}$$

where $S_i(k)$ represents the current state of the automaton user i resides in the k-th step. Among these subordinate states, we use the superscript "+" to represent the one to the right-hand side, and "−" for the one to the left-hand side. Formally, we have:

$$\sigma_j = \begin{cases} \left\{0_+^1, 0_+^2, \ldots, 0_+^{\frac{\beta}{2}}\right\}, & \text{if } j = 0, \\ \left\{u_{\text{max,-}}^{\frac{\beta}{2}}, u_{\text{max,-}}^{\frac{\beta}{2}-1}, \ldots, u_{\text{max,-}}^{1}\right\}, & \text{if } j = u_{\text{max}}, \\ \left\{j_-^{\frac{\beta}{2}}, j_-^{\frac{\beta}{2}-1}, \ldots, j_-^1, j_+^1, j_+^2, \ldots, j_+^{\frac{\beta}{2}}\right\}, & \text{otherwise,} \end{cases}$$

where β is introduced as a non-negative even number. That is,

- state 0 has $\frac{\beta}{2}$ transitional states on its right;
- state u_{max} has $\frac{\beta}{2}$ transitional states on its left;
- other steady states have β subordinate states, where $\frac{\beta}{2}$ of them to its left, and $\frac{\beta}{2}$ states to its right.

Therefore, the total number of transitional states is:

$$\sum_{j=0}^{u_{\text{max}}} |\sigma_j| = (u_{\text{max}} + 1 - 2)\beta + 2 \times \frac{\beta}{2} = \beta u_{\text{max}}. \tag{18}$$

Fig. 3 shows an example of our proposed Gur Game structure, where $u_{\text{max}} = 2$, i.e., $j = 0, 1, 2$ and $\beta = 2$, i.e., each steady state has two transitional states on both sides.

C. QoI index

To describe the level of satisfaction of a QoI attribute after receiving the feedback from participants, we define the "QoI index" for each step k as:

$$I(k) \triangleq \tanh\Big(\eta \ln \frac{u^a(k)}{u^r}\Big), \tag{19}$$

where $u^a(k) = \sum_{i=1}^N u_i^a(k) = \sum_i p_i u_i(k)$, and η denotes a scaling factor that determines the "range" of the values for the ratio of the attained and required QoI. The selection of the functions $\ln(\cdot)$ and $\tanh(\cdot)$ is rather arbitrary, but results in the intuitively appealing and desirable behavior for satisfaction. If $u^a(k)$ is far more than u^r, $I(k)$ will be infinitely close to 1. If $u^a(k)$ is far less than u^r, $I(k)$ will be infinitely close to -1. The condition of $I < 0$ indicates that attained QoI is still insufficient, while $I > 0$ for over-satisfaction. Actually, the QoI index defined in (19) behaves symmetrically around the origin, rising from -1 to 1, with the value 0 signifying the case where the QoI expectations are exactly satisfied, namely $u^a(k) = u^r$. In this way, the QoI index quantitatively describes the level of satisfaction of a QoI attribute, given the attained and required QoI.

D. Energy-Aware Pay-off Structure

Let $R_i(k)$ and $P_i(k)$, where $k = \{1, 2, \ldots, K\}$, denote the reward and penalty user i receives at the k-th step of iteration, respectively. Then, the current state of the automaton will transit probabilistically according to the received, collective pay-off value from all participants. From Fig. 3, under the reward, we observe that the steady states will stay unchanged, while their governed transitional states shift to themselves. On the contrary, the penalty will drive the opposite direction of state shifting. In a summary, it is interesting to see that the reward motivates the automaton to shift to/stay at the steady states, while the penalty causes the automaton to leave from the steady states and swing among the transitional states.

The basic goal for pay-off structure design in Gur Game is, when the attained QoI is largely satisfied, the reward probability R should be higher enough to keep the automaton state stable. However, if the attained QoI is not sufficient or excessive, the penalty probability P should be higher to stimulate state transition, gradually converging to desired states. Since the recommended action of each user is different, so is the number of received answers $u^a(k)$, and the QoI index $I(k)$. Thus, we introduce a novel reward structure for the k-th iteration user i receives. It considers both the attained QoI index $I(k-1)$ in the previous step, and its corresponding action j, as:

$$R_i^j(k) = g\Big(I(k-1), j\Big), \quad \forall i, j, k. \tag{20}$$

Function g: $\mathbb{R}^2 \to [0, 1]$ denotes the mapping from two inputs to the reward probability. Then, the corresponding penalty structure is simply as:

$$P_i^j(k) = 1 - R_i^j(k), \quad \forall i, j, k. \tag{21}$$

An example of $R_i^j(k)$ and will be used in our evaluation is:

$$R_i^j(k) = \begin{cases} e^{-\rho_j I(k-1)^2}, & \text{if } I(k-1) \in [0, 1), \\ e^{-\tau_j I(k-1)^2}, & \text{otherwise.} \end{cases} \tag{22}$$

Note that (22) exactly implements the basic goal for pay-off structure design. Fig. 4 shows this implementation, where parameters are set as $u_{\text{max}} = 2, \eta = 2$. First, from the discussion in Section V-C, we can see that both insufficient and superfluous information contribution will lead to a lower reward probability. Second, to shape the state transition more precisely, we consider the difference of individual action by introducing parameter ρ and τ. In this example, we set $\rho_0 = \tau_2 = 2\alpha, \rho_1 = \tau_1 = 5\alpha, \rho_2 = \tau_0 = 8\alpha$. In other words, at any iteration step, we assign different reward functions to different actions (i.e., the number of answers denoted by j), even though they receive the same achieved QoI index. That is, when $I < 0$, higher j is assigned a higher reward than

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

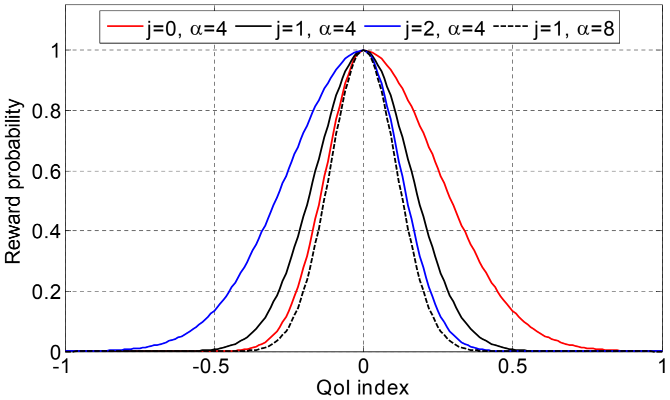

Fig. 4. An example of the reward probability for different j and α.

the states with smaller j; and when $I > 0$, the situation is opposite. This is because that when the attained QoI is insufficient (or: $I < 0$), it is more preferable to have the participant's automaton to shift to the right-hand side states (associated with bigger j) with a higher reward. Therefore, compared with users with smaller recommended j, users with bigger j should be assigned higher reward to keep it stay at the current states, rather than shifting to the left. However, when the attained QoI is over-satisfied, it is reasonable to motivate users associated with higher j to shift to the left (thus lower reward). Collectively, the effects of (22) is to recommend user actions to achieve the exact required QoI (or $I = 0$), and the participants in the Gur Game can collaboratively achieve the highest reward probability through limited steps of iterations. Furthermore, parameter α adjusts the breadth of the reward function. Higher α makes the shape of reward function more concentrated, as shown in Fig. 4 where α changes from 4 to 8 of the same $j = 1$.

E. Two-Step Decision-Making Process

Given the pay-off structure, we next show the proposed iterative and distributed decision-making process for each user i. Recall that the goal of our participatory crowdsourcing framework is to find an optimal vector $\{u_i\}^*$ as the number of answers participants are recommended to respond, which also achieves a satisfactory level of energy consumption fairness among all participants. To solve this, we propose a *two-step* Gur Game operation. QoI step (or Q-step) aids to regulate the collectively achieved QoI within a small range just above the required value, i.e., $[u^r, (1+\varsigma)u^r]$, where $\varsigma \in [0, 1]$. Then, the variance step (or V-step) minimizes the variance of the total energy consumption ratio to provide the maximum extent of fairness. Note that our proposed solution is purely distributed, and thus the following descriptions apply for every user i.

Q-step: At the k-th iteration, if $u^a(k-1) - u^r < 0$ or $u^a(k-1) - u^r > \varsigma u^r$, the automaton transits its state probabilistically according to the received pay-off value.

V-step: Denote the left/right adjacent state of $S_i(k)$ as $\mathsf{L}(S_i(k))$/$\mathsf{R}(S_i(k))$. At the k-th iteration, if the V-step criterion $0 \le u^a(k-1) - u^r \le \varsigma u^r$ satisfies, the automaton transits its current state either to its left or right-hand side, according to the comparison between its own energy consumption ratio

Algorithm 1 Gur Game-based Response Recommendation

set $u_i(0) = u_{\max}, S_i(0) = u_{\max}, \forall i \in \mathcal{N}$, and $I(0) = 1$.
for all $k = 1, 2, 3, \ldots K$ **do**
  compute $u^a(k-1)$ in (2);
  if $u^a(k-1) - u^r < 0$ or $u^a(k-1) - u^r > \varsigma u^r$ **then**
    compute $I(k-1)$ in (19);
    for all user $i \in \mathcal{N}$, **do**
      compute $R_i^j(k)$ and $P_i^j(k)$ in (20) and (21);
      generate a random number `seed`$\in [0, 1]$;
      if `seed` $\le R_i^j(k)$ **then**
        transit state by the reward behavior;
      else
        transit state by the penalty behavior;
      end if
    end for
  else
    transit state by (23);
  end if
end for
find the final solution $\{u_i\}^*$ by (24) or (25).

and the average value among all participants. Let $\bar{\xi}(k-1) = \frac{1}{N}\sum_{i=1}^{N} \xi_i(k-1)$ denote this average value, and $\epsilon \in [0, 1]$ as an adjustment threshold. Specifically, we have

$$\begin{cases} S_i(k) \to \mathsf{L}(S_i(k-1)), & \text{if } \frac{\xi_i(k-1)-\bar{\xi}(k-1)}{\bar{\xi}(k-1)} > \epsilon, \\ S_i(k) \to \mathsf{R}(S_i(k-1)), & \text{if } \frac{\xi_i(k-1)-\bar{\xi}(k-1)}{\bar{\xi}(k-1)} < -\epsilon, \\ S_i(k) \to S_i(k-1), & \text{others.} \end{cases} \quad (23)$$

From (23), we observe that if ξ_i exceeds the average to a certain extent, the automaton state shifts to the left, yielding an action of decreasing number of answers, and vice versa. For completeness, if $S_i(k) = 0$ or $S_i(k) = u_{\max}$, we cannot shift it to the left or to right. In this situation, we simply keep the current state unchanged.

It is worth noting that in practice the V-step and Q-step alternate, through the trial-and-error based on the obtained QoI. The k-th iteration is associated with a vector $\{u_i(k)\}$ that corresponds to a recommendation of returned answers, achieved QoI and variance of ξ_i. When the K-th iteration completes, we seek for a suboptimal solution from all these iteration steps, based on the following criterion:

1) if $\forall k = 1, 2, \ldots, K$, there exits at least one $u^a(k) \in [u^r, (1+\varsigma)u^r]$, then we find the iteration step k^* that achieves the minimum variance of $\xi_i, \forall i$:
$$k^* = \text{argmin}_k \; V(k), \text{if } u^a(k) \in [u^r, (1+\varsigma)u^r], \quad (24)$$
and the final solution is given by $\{u_i\}^* = \{u_i(k^*)\}$.
2) otherwise, we select the iteration step k^* to achieve the QoI with minimum increment to $(1+\varsigma)u^r$:
$$k^* = \text{argmin}_k \left(u^a(k) - u^r\right), \; \forall u^a(k) \in (u^r, \infty), \quad (25)$$
and the final solution is given by $\{u_i\}^* = \{u_i(k^*)\}$.

The solution is given by computational complexity $O(K)$.

The pseudo-code in Algorithm 1 illustrates the proposed two-step Gur Game control. It is worth noting that the our proposed approach is fully distributed. Users neither need to forecast their own energy-consumption states nor exchange any information with other participants. Instead, they use

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

the way of trial-and-error to produce the best result at each step and iteratively achieve the overall suboptimum, which is confirmed in the simulation section.

VI. Incentive-Based Participant Selection

One problem of using participatory crowdsourcing for information gathering and retrieval is how to effectively motivate the users' participation. In reverse auctions based approaches [25], users first bid for selling their sensory data, and then the service provider selects predefined number of users with lowest bids. The selected users receive their bidding prices as a reward. Since the bid is decided by participants, it simplifies the pricing decision from the platform's point of view and users join the competition between others as if they are playing a game. Similar to [25], we utilize reverse auction as the main framework of our incentive-base participant selection mechanism, but we explicitly consider the changing bid of a user during a few consecutive crowdsourcing tasks, by modeling the user bidding behavior.

Let $b_i, \forall i \in \mathcal{N}$ denote the bid of user i, representing the expected paid credits for unit amount of information contribution. After the previous interaction between the network platform and user's smart device, the platform is notified the preliminary action of each user, namely, the information contribution determined by the Gur Game engine. Next, users send their bids to the platform. A primitive way is to recruit all N users and reward them all as their wish, and we use this approach as a "benchmark". Intuitively, users behave aggressively to gain credits as much as possible, and this primarily drives the need of investigating enhancements to minimize the total paid credits from the network platform. This section deals with the following problem: *from the network platform's perspective, given the presumed user behaviors, how to minimize the total paid credits to participants, and achieve the QoI requirements of the questionnaire request simultaneously?*

We propose a heuristic user selection and credit allocation algorithm to tackle this challenge. First, we relax the required QoI in (7) as $\tilde{u}^r = (1+\delta)u^r$, where δ is a tunable margin to let the Gur Game engine distributely provide superfluous information contribution for the request, and $\delta > 0$. Then, according to the received user bid $b_i, \forall i \in \mathcal{N}$, the platform "removes" the participants in a descending order of b_i until the collective contribution $\tilde{u}^a$ approaches u^r with minimum increment. That is, we reject users with high bidding prices. We use the set $\mathcal{M}$ with size M to denote the remaining participants who are finally selected to contribute information and receive rewarding credits. The pseudo-code in Algorithm 2 illustrates this process.

Let φ and $\tilde{\varphi}$ denote the total paid credits by the benchmark and proposed approach. We have

$$\varphi = \sum_{i=1}^{N} b_i p_i u_i, \tag{26}$$

and

$$\tilde{\varphi} = \sum_{i=1}^{M} b_i p_i \tilde{u}_i, \tag{27}$$

Algorithm 2 User Selection and Credit Allocation

```
set M = N, ũ^r = (1+δ)u^r, replace the u^r in Algorithm 1 with ũ^r;
run Algorithm 1, get {ũ_i}*, ũ^a = Σ_{i=1}^N p_i ũ_i, ∀i ∈ N;
sort users according to their bids, b_1 ≤ b_2 ≤ ··· ≤ b_N;
for all i = N, N−1, N−2, ..., 1 do
    if ũ^a − p_i ũ_i ≥ u^r then
        ũ^a = ũ^a − p_i ũ_i;
        M = M − {i};
    else
        break;
    end if
end for
use {ũ_i | ∀i ∈ M}* as the selection result;
allocate credit b_i to user i, ∀i ∈ M.
```

where

$$\sum_{i=1}^{N} p_i u_i = u^r, \; \sum_{i=1}^{N} p_i \tilde{u}_i = \tilde{u}^r = (1+\delta)u^r, \\ b_1 \le b_2 \le \cdots \le b_M \le b_{M+1} \cdots \le b_N. \tag{28}$$

Theorem 6.1: Compared with the benchmark, the unnecessary and sufficient condition for credits saving by our proposed approach is given by

$$\theta < \frac{\sum_{i=1}^{M} p_i (b_{M+1} - b_i)(\tilde{u}_i - u_i)}{b_{M+1} u^r}, \tag{29}$$

where $\theta = \sum_{i=1}^{M} p_i \tilde{u}_i / u^r - 1$ is a judging parameter.

Proof: From Algorithm 2, it is clear to see that $\sum_{i=1}^{M} p_i \tilde{u}_i \ge u^r$. Then, we introduce a non-negative parameter θ where $\sum_{i=1}^{M} p_i \tilde{u}_i = (1+\theta)u^r$. Subtracting (27) from (26) and given (28), we have:

$$\begin{aligned} \varphi - \tilde{\varphi} &= -b_1 \theta u^r + \sum_{i=2}^{M} p_i (b_i - b_1)(u_i - \tilde{u}_i) \\ &+ \sum_{i=M+1}^{N} (b_i - b_1) p_i u_i \\ &\ge -b_1 \theta u^r + \sum_{i=2}^{M} p_i (b_i - b_1)(u_i - \tilde{u}_i) \\ &+ (b_{M+1} - b_1) \sum_{i=M+1}^{N} p_i u_i. \end{aligned} \tag{30}$$

By removing the users with higher bid, we reduce the collective contribution towards u^r. Note that $\sum_{i=1}^{N} p_i u_i = \sum_{i=1}^{M} p_i \tilde{u}_i - \theta u^r$. Hence, $\sum_{i=M+1}^{N} p_i u_i = \sum_{i=1}^{M} p_i (\tilde{u}_i - u_i) - \theta u^r$. Putting this back to (30), finally we obtain:

$$\varphi - \tilde{\varphi} \ge \sum_{i=1}^{M} p_i (b_{M+1} - b_i)(\tilde{u}_i - u_i) - \theta b_{M+1} u^r. \tag{31}$$

Therefore, if (29) is satisfied, then $\varphi - \tilde{\varphi} > 0$. This completes the proof. ■

Next, we give a thorough analysis on Theorem 6.1. Since we introduce a margin δ to produce superfluous preliminary

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

information contribution, we have $\tilde{u}_i \geq u_i$, $\forall i \in \mathcal{N}$,[1] and consequently, $\sum_{i=1}^{N} b_i p_i \tilde{u}_i > \sum_{i=1}^{N} b_i p_i u_i$. Clearly, when a participant is removed, the total credits are decreased, until the algorithm ends by finding the minimum QoI increment upon u^r, and this closeness is characterized by parameter θ in (29). If we exactly reduce the provided QoI to the required value u^r, i.e., $\theta = 0$, since $\sum_{i=1}^{M}(b_{M+1} - b_i)p_i(\tilde{u}_i - u_i)$ is positive[2], we have $\varphi - \tilde{\varphi} > 0$. However in most cases, since the final produced QoI is not exactly u^r, whether or not we achieve in saving credits depend on the judging parameter θ in Theorem 6.1.

Moreover, we explicitly consider different user behavior on credit acquisition. For each questionnaire request, the platform executes a round of reverse auction to select users. For users who win the current auction, they may increase their bidding prices for future tasks to maximize their expected profits; while for losers, they may decrease their expectations and lower their future bids accordingly. Therefore, we model the user's behavior as:

1) adaptive bidding with proportional change:

$$b_i = \begin{cases} (1+\kappa)b_i, & i \in \mathcal{M}, \\ (1-\kappa)b_i, & i \notin \mathcal{M}, \end{cases} \tag{32}$$

where κ denotes the ratio to signify the bidding change in consecutive crowdsourcing tasks, and $\kappa > 0$.

2) adaptive bidding with fixed change:

$$b_i = \begin{cases} b_i + \chi, & i \in \mathcal{M}, \\ b_i - \chi, & i \notin \mathcal{M}, \end{cases} \tag{33}$$

where χ is the step size for bid change, and $\chi > 0$.

It is worth noting that if the participants' bids are close enough, the right-hand side of (29) would be relatively low, making it difficult to satisfy (29); while if the variance of their bids is large enough, the the right-hand side of (29) would be relatively higher, having more opportunity to achieve credit saving. We will confirm this analysis in Section VII-C.

VII. Performance Evaluation

A. Setup

We compare our proposed Gur Game based approach with the "min-answers" method that achieves the required QoI while minimizing the total number of answers, irrespective the energy consumption of different users, as:

$$\begin{aligned} \textit{minimize:}_{\{u_i\}} \quad & \sum_{i=1}^{N} u_i, \\ \textit{subject to:} \quad & u^a \geq u^r, \\ & 0 \leq u_i \leq u_{\max},\ u_i \in \mathbb{Z}, \quad \forall i \in \mathcal{N}, \end{aligned} \tag{34}$$

and the solution is given by [31]. In addition, since our proposed Gur Game based approach is a heuristic algorithm and offers a suboptimum for (7), we also compare with the centralized, optimal solution given by Theorem 5.1.

[1] It is worth noting that both $\{u_i\}^*$ and $\{\tilde{u}_i\}^*$, $\forall i \in \mathcal{N}$ are obtained by solving the optimization (7) with the objective of minimizing the corresponding variance V. The difference is that $\{\tilde{u}_i\}^*$ achieves the QoI of $(1+\delta)u^r$, which is larger than $\{u_i\}^*$ with the QoI of u^r. Consequently, it is obvious to conclude $\tilde{u}_i \geq u_i$, $\forall i \in \mathcal{N}$.

[2] From (28), we have $b_{M+1} \geq b_i$, $\forall i \in \{1, 2, \ldots, M\}$ and hence, $\sum_{i=1}^{M}(b_{M+1} - b_i)p_i(\tilde{u}_i - u_i) > 0$.

TABLE II
Simulation parameters

Parameter	Description
$N = 47$	total number of participants
$u^r = 148$	required QoI
$\gamma = 30$	normalization factor for ξ_i
$K = 500$	maximum iteration step
$u_{\max} = 6$	maximum allowed number of answers per user
$\beta = 2$	no. of subordinate states of each steady state
$\varsigma = 0.015$	the range for satisfactory QoI in Q-step
$\epsilon = 0.05$	the adjustment threshold in V-step
$\eta = 2$	scaling factor for QoI index
$\alpha = 8$	scaling factor for reward function

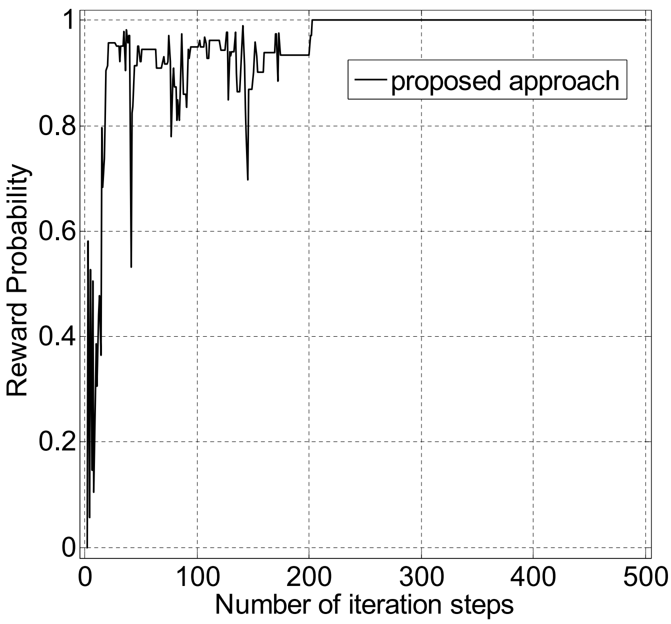

Fig. 5. Achieved reward *vs.* iteration steps.

Our simulation is based on the treatment and systematic assumptions described in Section IV, and we restate them as follows. We focus on the collection of a sufficient number of answers to the questionnaire of the student's health condition, on rating as "average". From the data, we observe 47 students eventually participate. We simulate the request every three days, during which we assume the questionnaire is delivered 6 times to the students. In total we have 91 time periods. On receiving the request, the smart device of each participant runs our proposed Gur Game automaton iteratively to make the recommendation on the number of returned answers. Then, the QoI index and reward probability are computed as (19) and (22), respectively. That is, since $u_{\max} = 6$, we have 7 steady states in each automaton, and they are associated with reward function parameters $\rho_j = \alpha(j+2), \tau_j = \alpha(u_{\max}+2-j)$. Other parameter settings are given in Table II. It should be pointed out that the Gur Game automaton can start at random initial state. For simplicity, we set the initial state $S_i(0) = u_{\max}$, and thus, the achieved QoI is expected to decrease from time-being.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

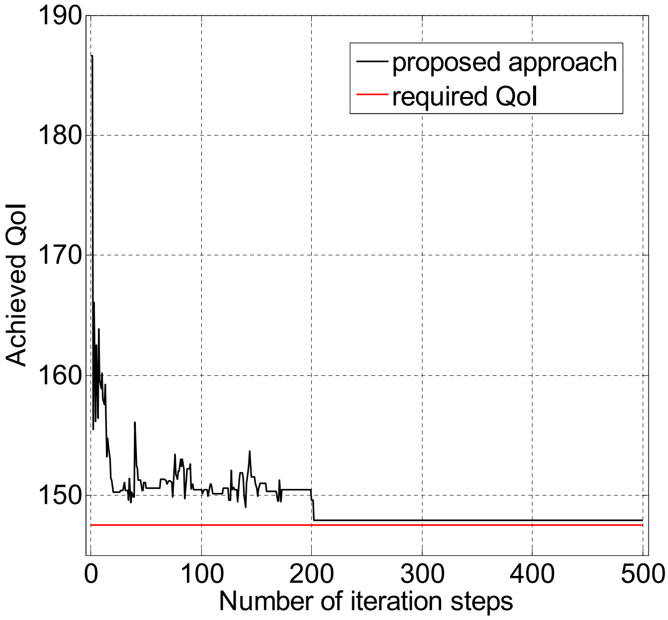

Fig. 6. Achieved QoI *vs.* iteration steps.

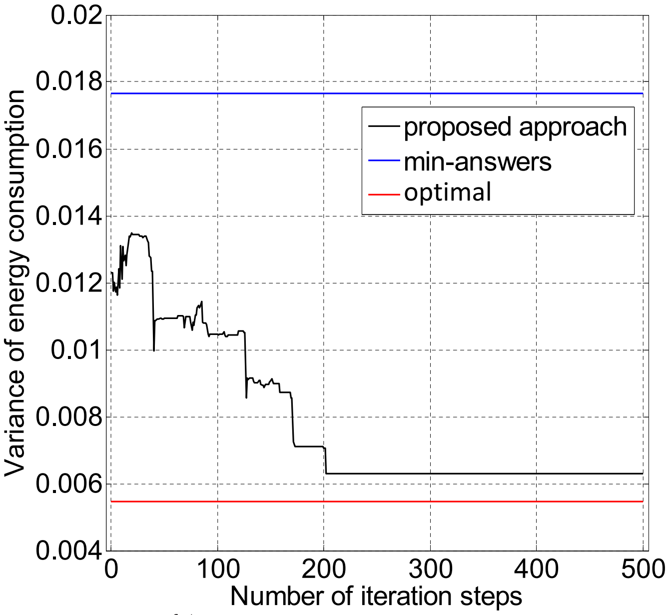

Fig. 7. Variance of ξ_i *vs.* iteration steps.

B. Convergence and System Performance

Without loss of generality, we first pick the 40-th time period and show the system convergence in term of the change of received reward probability, achieved QoI, the variance of $\xi_i, \forall i$ and total energy consumption in Fig. 5, Fig. 6, Fig. 7 and Fig. 8 respectively. When Gur Game begins, since we set the number of answers each user reports as the maximal, the achieved QoI far exceeds the required value, thus making the achieved reward very small. Gradually, the automaton of each user transits its state probabilistically according to the new pay-off value, and it successfully lowers the achieved QoI close enough to the required one. It is worth noting that the downhill (uphill) of the achieved QoI (see Fig. 6) corresponds to the uphill (downhill) of the reward probability (see Fig. 5). After only 200 steps, the reward is very close to 1, leading to the system convergence with negligible computational com-

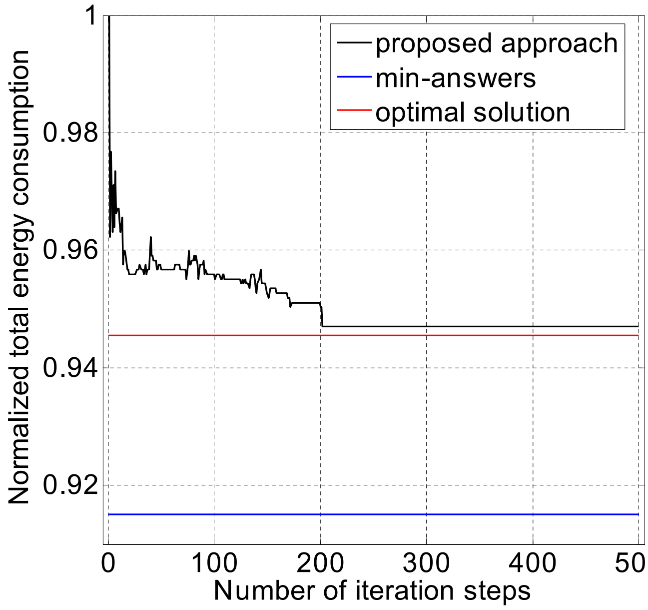

Fig. 8. Normalized total energy consumption *vs.* iteration steps.

pelxity.

The effect of our proposed two-step adjustment approach is clear when the achieved QoI falls in the range of $u^a \in [148, 150]$, i.e., the automaton finishes the Q-step and now enters the V-step for finer adjustment. In the V-step, each user's automaton state is shifted slightly either to the right or left-hand side of the current state which may cause the recommended action (the number of returned answers) to change slightly as well. However, as long as the collective total number of answers remains satisfactory (the criterion of Q-step), the energy consumption fairness can be gradually achieved in a distributed manner. This is confirmed in Fig. 7 as the tiny gap of the variance of $\xi_i, \forall i$ between our solution and the optimum. Moreover, we observe that the "min-answers" method cannot guarantee a satisfactory level of energy consumption fairness.

Compared with exhaustive search, our approach successfully lowers the computational complexity from $(u_{\max}+1)^N$ to just a few hundred steps of iterations. Note that for Gur Game, how long it takes to reach the optimal system state depends on a number of system parameters, like the population size N and the reward function. The larger number of users, smaller reward changing step, longer the iteration would take. The proposed Gur Game-based framework consumes very limited additional bandwidth and energy during convergence.As shown in Fig. 1, the piece of information exchanged between the base station and smart devices only includes the properties of that information and the pay-off value, but not the actual multimedia data. Practically, in our case study we need only the meta data like task ID (32bits), the user ID (32bits) and the number of answers (32bits), but not the questionnaire itself (thus in total 96bits). To further verify this, we vary the number of users, downscale the required QoI proportionally and show the required iteration steps for convergence, and total amount of exchanged information (including overhead), as shown in Fig. 9. It is not surprising to see that our

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

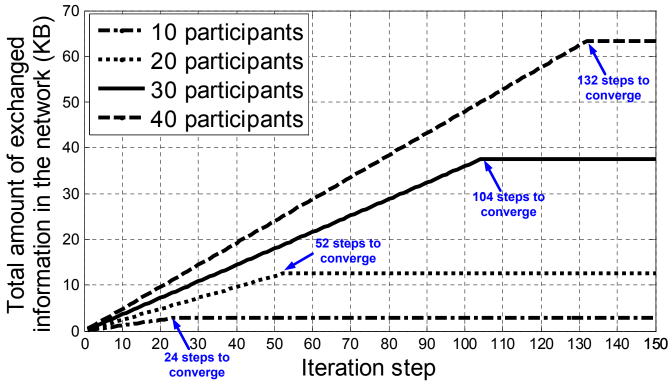

Fig. 9. Number of iteration steps required for convergence and total exchanged information.

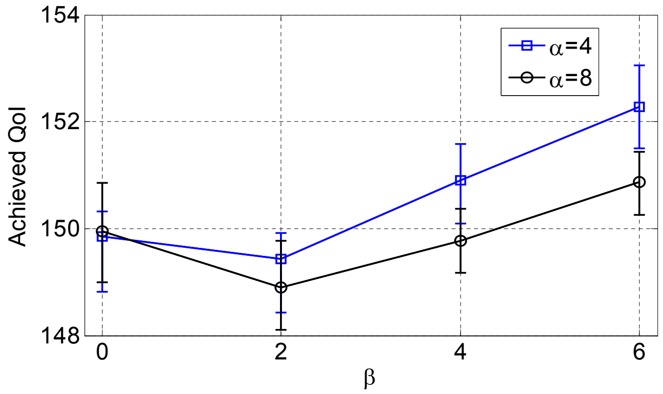

Fig. 10. Achieved QoI for the 40-th time period under different β and α, with a total of 100 runs.

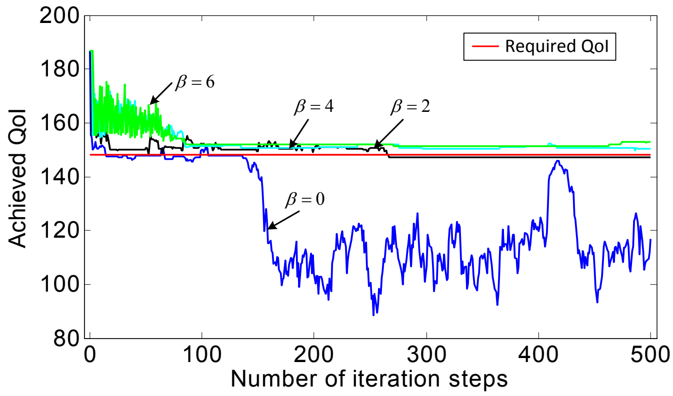

Fig. 11. The change of achieved QoI under different number of transitional states, when $\alpha = 8$.

approach achieves linear performance when time progresses, since every round of iteration is independent. With the increase of N, although more steps are required, the advantage of Gur Game still cannot be neglected, since exhaustive search cannot be applied when N is large. Furthermore, compared with traditional mathematical optimization solving techniques, Gur Game has its flexibility and universality when the problem changes, since it is transparent to the specific form of the objective function. We can also observe that when $N = 40$ (as a reasonable crowd of users), total aggregated information for the considered application is only 62KB, which will not cause bandwidth overuse. Therefore, we can safely conclude that although Gur Game needs feedback collection during many iteration steps, the piece of information exchanged per step is relatively small and can be easily piggybacked in the existing signaling payload (such as the regular paging/registration process in cellular networks). To this end, the feature of low bandwidth utilization and signaling overhead makes our approach a suitable solution for crowdsourcing network.

Fig. 8 compares the total energy consumption among three methods, where the value after 200 steps denotes the total energy consumption of the proposed approach. It is obvious that "min-answers" method spends less energy, while the optimal solution (in minimizing the variance) and our proposed approach consume a bit more. Combining this result with Fig. 7, we observe that the energy saving of the "min-answers" is at the expense of sacrificing the user fairness, since it neglects the residual energy state of different users. However, both the optimal solution and our proposed approach aim to guarantee a satisfactory level of fairness in user participation, and thus they well balance the information contribution with user device's energy reserve. From QoI perspectives, what it matters is whether the collocated participatory crowdsourcing network can sustain the future tasks without early dying nodes. Therefore, in our study we consider the user participation fairness as the objective function.

Next, we explore the system behavior under different Gur Game automaton settings, and investigate the stability of our proposed approach. Recall that β determines the number of transitional states in the Gur Game structure, and α influences the shape of reward function, and these two parameters together impact on the speed of system convergence as evaluated next. We perform 100 Monte-Carlo runs on the 40-th time period. Fig. 10 demonstrates the achieved QoI with 90% confidence interval, varying α, β and keeping the other parameters as basic settings. When $\alpha = 8, \beta = 2$, 90% confidence interval of the achieved QoI is within range [148.10, 149.67], as the best performance among all tested combinations. To illustrate the effect of α, β, we zoom in one result from the total of 100 repeats. By varying β in Fig. 11, we see that if no transitional states are employed, i.e., $\beta = 0$, the QoI does not converge even with the maximum iteration step. This is because without the "buffering" effect of the transitional states, the state of Gur Game automaton changes fiercely and recklessly in the V-step, where even a minor state

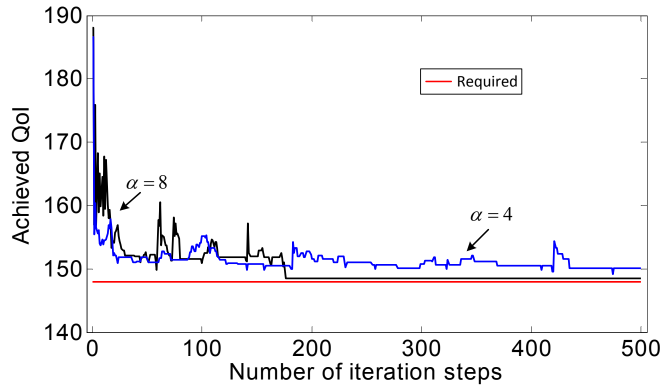

Fig. 12. The change of achieved QoI under reward functions with different α, when $\beta = 2$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

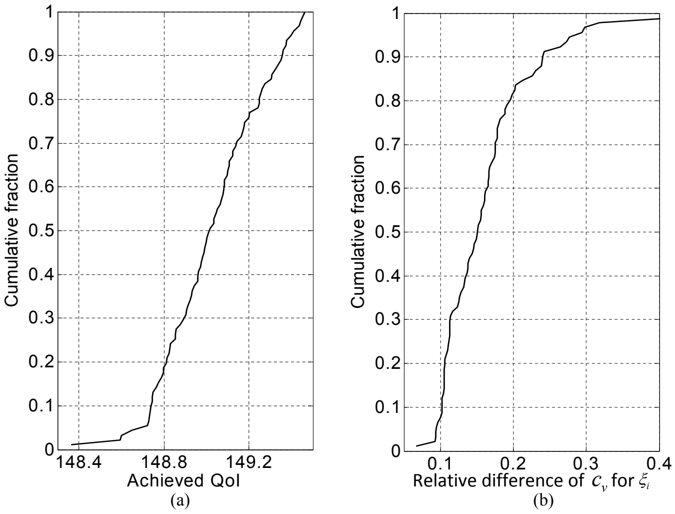

Fig. 13. Cumulative fraction of (a) the achieved QoI, (b) relative difference of c_v for ξ_i, for total 91 time periods.

change of each user's automaton from the current one to the adjacent state (representing different number of answers) can significantly cause the achieved QoI to change dramatically, and in the worst case the automaton has to go back to the Q-step due to the small reward value. Therefore, the system is very difficult to stay stable, nor converge. This effect is confirmed when $\beta = 4, \beta = 6$, where the achieved QoI is still out of the adjustment range of V-step even after the maximum iteration step. Towards this end, we may conclude that $\beta = 2$, or two transitional states associated with each steady state is the optimal configuration.

Fig. 12 show the effect of α when $\beta = 2$ to illustrate the impact of the breadth of reward function on system convergence. Apparently, the wider the reward function (e.g., $\alpha = 4$ compared to $\alpha = 8$), the slower the change of reward with respect to the same level of QoI index change (see Fig. 4), and thus the slower the speed of system convergence. In other words, given the same received QoI index, the concentrated reward function represents more stringent requirement for QoI satisfaction (lower reward value compared with what the wider function produces), and as a result, the trend to the desired QoI ($I = 0$) is more signified and eventually reflected by the faster system convergence.

We perform Monte-Carlo runs on total 91 time periods and compute the relative coefficient of variation c_v for the total energy consumption ratio among all participants. Furthermore, we compute, to what extent the obtained c_v (by proposed Gur Game approach) approaches the optimality, shown as its relative difference in Fig. 13, together with the achieved QoI. In Fig. 13 we observe that the achieved QoI of all tests is above the required value with a small interval, and among 90% tests the relative difference of c_v is lower than 25%. These results indicate a well-acceptable performance of our proposed approach.

Finally, to show the adaptability of our solution towards different information quality models, we use the information fusion function in (6) which captures decreasing marginal returns in the amount of information, and show its system convergence in Fig. 14 and Fig. 15. The parameter is $\vartheta = 0.98, \varpi = 100$. Clearly, the evolving trend of achieved QoI is the same with Fig. 6, where it successfully approaches the required QoI after a few iteration steps. Fig. 15 shows that the energy consumption fairness can also be gradually achieved. This is because in Gur Game, no matter how a user voted, the user independently transits its state according to the reward/penalty probability calculated from the collective behavior of all users (the total information contribution in our case). After enough trails, the automata will reach the desired states, where the reward function reaches its peak, and this property holds no matter what characteristics the function has [21]. In fact, the individual automata know neither the reward function nor the information fusion function. Moreover, if the fusion function changes, the automata can adapt themselves automatically to the new function. So, the framework of Gur Game allows our system to react dynamically to the variations in real situations.

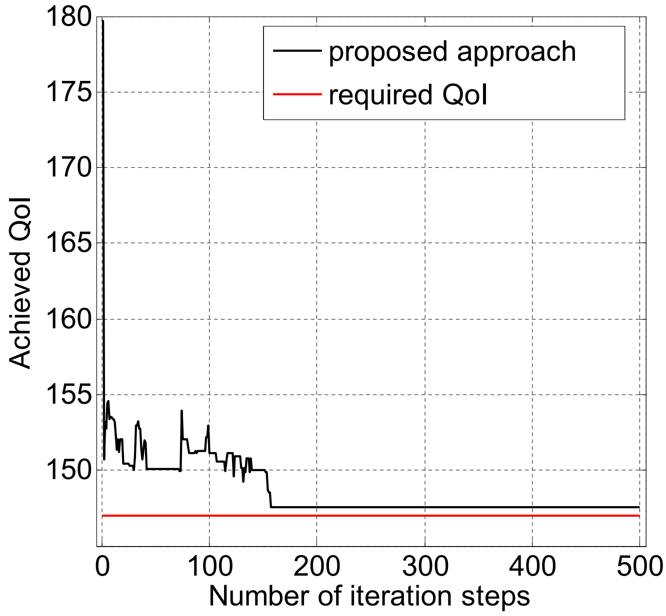

Fig. 14. Achieved QoI *vs.* iteration steps, with fusion function in (6).

C. Proposed Incentive-based Participant Selection Scheme

Based on parameter settings in Table II, we conduct a series of simulations to investigate the performance of proposed user selection and credit allocation approach in Section VI. We assume that the platform holds an upper bound for bidding price as $b_i \in [0, 30]$, $\forall i \in \mathcal{N}$, and we will show its effect in limiting the aggressive behaviors of the participants during the bidding process.

Without loss of generality, we perform 100 runs for both the proposed "fixed bidding" (i.e., all participants do not change the bid for a series of tasks) and benchmark approaches on the 40-th time period. For each run, we randomly generate initial bids for all participants, following the uniform distribution over [0,30]. Fig. 16 shows the achieved average credit saving, where we observe that proposed approach successfully reduces the total paid credits, and this gain becomes larger with higher δ. This is because higher δ (i.e., higher QoI requirement) drives the participants to contribute more and thus providing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

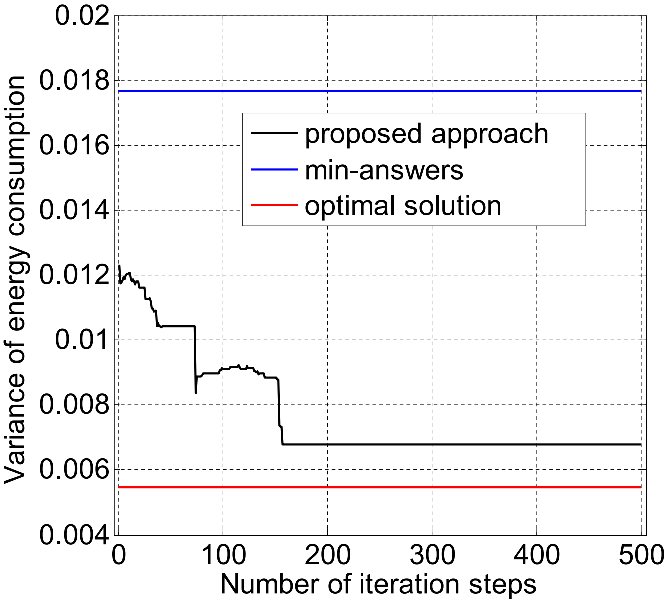

Fig. 15. Variance of ξ_i *vs.* iteration steps, with fusion function in (6).

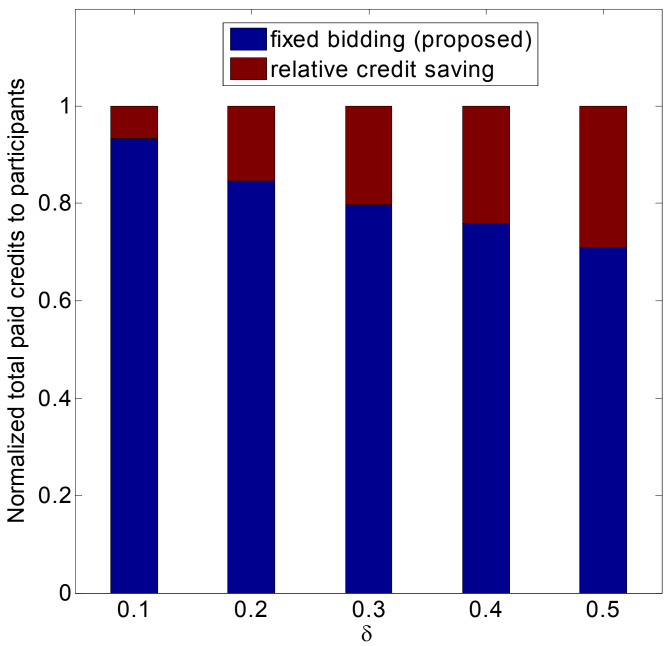

Fig. 16. Normalized total paid credits to participants *vs.* margin δ, when simulating the 40-th time period.

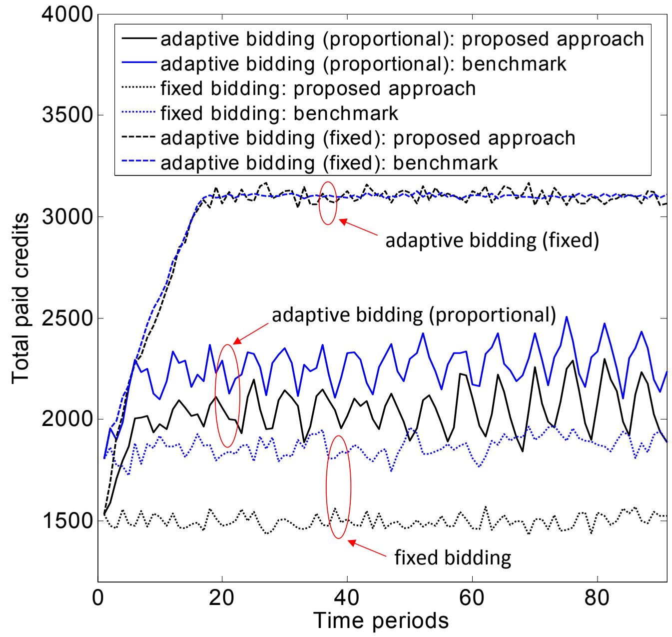

Fig. 17. Total paid credits *vs.* time periods, when $\delta = 0.5$, $\kappa = 0.4$, $\chi = 2$.

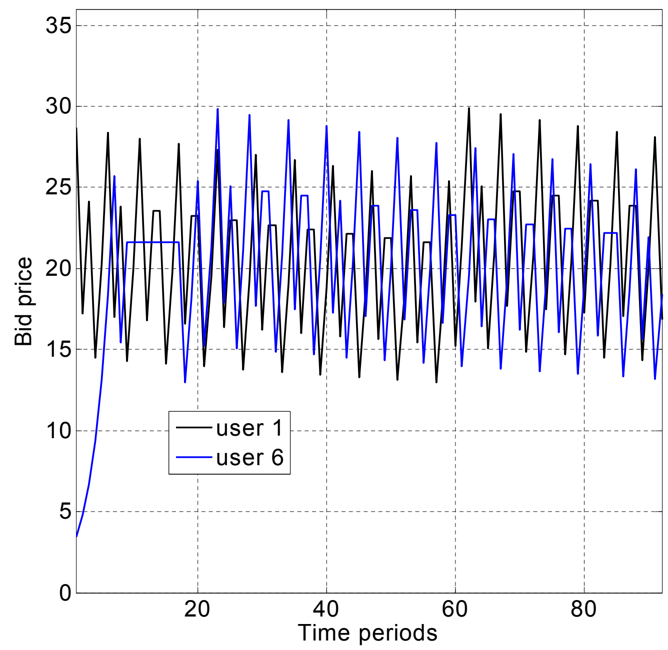

Fig. 18. The change of user bidding price in adaptive bidding with proportional change.

a higher degree of flexibility for the platform to choose the most "appropriate" participants with not only higher bids and smaller amount of contributed information to be removed from the user selection phase. This in turn rejects greedy users.

We next demonstrate the adaptive bidding process in Fig. 17 from time domain over 91 time periods (i.e., participants change their bid based on the previous auction result). We set $\delta = 0.5$ and compare the proposed approach with benchmark under three different settings, namely: (a) adaptive bidding with proportional change (referred as "proportional"), (b) adaptive bidding with fixed change (referred as "fixed"), and (c) fixed bidding. For fixed bidding, we also randomly initialize the bidding prices of all participants, and keep them constant during 91 time periods. For setting (a), we set a relatively large adaptation ratio $\kappa = 0.4$; for setting (b), we set a small step size $\chi = 2$, to demonstrate its impact on credit saving, as condition (29) explains. Apparently, we see two adaptive approaches dramatically lower the amount of paid credits compared with the benchmark, consistent with Fig. 16. Although initial bids are different, participants tend to progressively and aggressively raise their bids (see user 6 in Fig. 18 as an example) and thus decreasing their differences over time. This interesting observation shows their aggressive behaviors which eventually make them non-selective, and the impact can be more severe for the change with smaller step size $\chi = 2$. Sometimes it even fails to satisfy condition

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

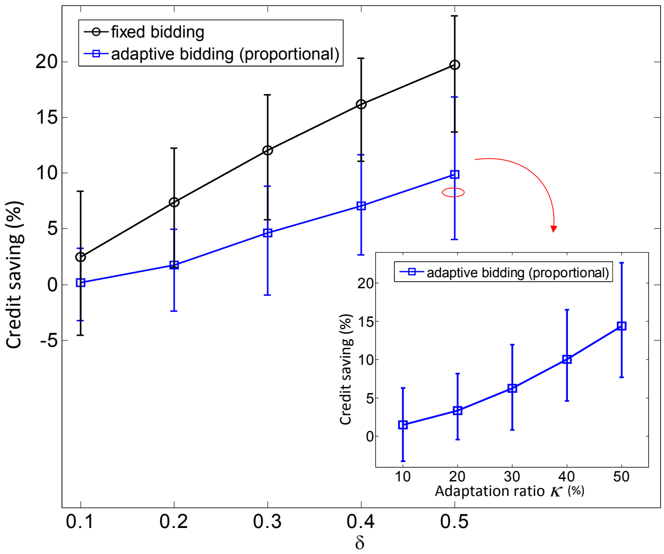

Fig. 19. Credit saving for total 91 time periods, varying margin δ and adaptation ratio κ.

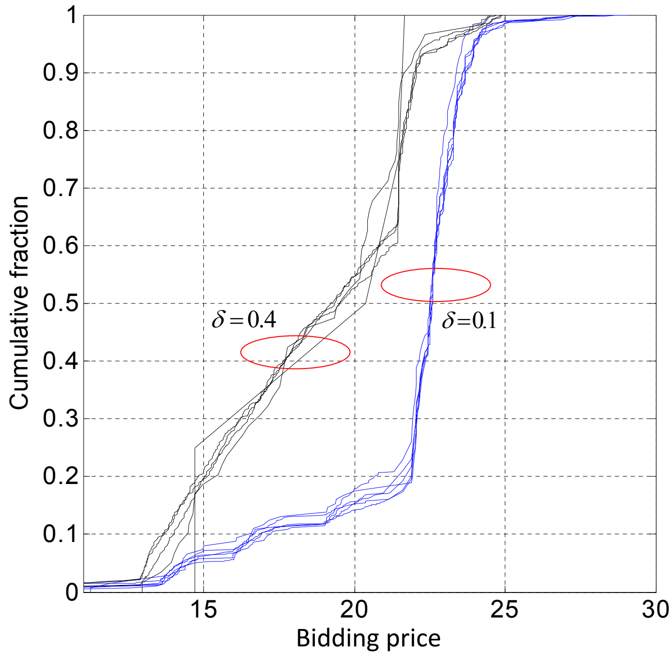

Fig. 20. Cumulative fraction of bidding prices for adaptive bidding (proportional) for total 91 time periods, when $\delta = 0.1$ and $\delta = 0.4$.

(29), thus potentially losing the benefit of credit saving. As confirmed in Fig. 17, two lines are interleaving for "adaptive bidding (fixed)". Furthermore, although all users are aggressively raising their bids, the upper bound of bidding price successfully limits this greediness, as shown in Fig. 17. This implies a practical system parameter setting to control the revenue by the platform. For "adaptive bidding (proportional)" scheme, it is interesting to observe that after the initially rapid growth, total paid credits fluctuate. This can be explained when zooming into randomly picked-up two users; see Fig. 18. Due to the characteristic of *proportional* change, if a user keeps losing-winning auctions successively, the bid itself is actually decreasing; only a series of consecutive wins can increase the bid. Hence, the collective effect of all participants similar to Fig. 18 eventually produces the shape in Fig. 17.

Fig. 19 confirms the results of Fig. 17 from another angle. Here we focus on "adaptive bidding (proportional)" by changing δ and κ. Varying $\delta = [0.1, 0.5]$ and setting $\kappa = 0.4$, we observe that when δ is small, the average amount of credit saving is also small; for some cases, the proposed approach even has more paid credits than the benchmark, thus confirming the conditional saving feature by Theorem 6.1. Next, we fix $\delta = 0.5$ and vary different κ to show the impact of different user behaviors. Clearly, our approach performs better in a more dynamic bidding environment, where the relatively higher adaptation ratio κ amplifies the user differentiations, thus increasing the benefits of user selection.

Finally, we explore the distribution of user bids over 91 time periods by simulating the "adaptive bidding (proportional)" algorithm. We show the cumulative fraction of bidding prices for 6 voting options, i.e., in our simulation, we set $u_i = \{1, 2, \ldots, 6\}$. Therefore, for different δ, we obtain 6 curves each, as shown in Fig. 20. When $\delta = 0.1$, the bidding prices are quite concentrated in a small range of [22, 25]. Again, this is due to the user's *aggressive* behavior to constantly raise bids after a successful auction. Contrarily, when $\delta = 0.4$, greedy users with high bidding prices can be effectively rejected by the platform, resulting in a much wider and lower range of bidding prices as shown in the figure.

VIII. Discussions

In this section, we discuss some practical issues of the proposed Gur Game-based approach.

A. Distributed Characteristics of the Gur Game

For a fully distributed control system, one aims to let users perform a cooperative task without outside control. Strictly speaking, the Gur Game-based approach is not a fully distributed control approach, since the contributed information needs to be collected by a central server (which is the base station in the considered cellular networks); however it concerns more about optimizations. In Gur Game, the automaton of each user independently transits its state, taking into account the feedback that captures the collective actions of all users. By introducing the base station as the coordinator, there is no need to exchange information between users. Moreover, the base station acts not as a commander, but more like a globally observable quantity [22].

From a practical point of view, this feature of our Gur Game-based approach exactly fits for the scenario of existing cellular networks, since there exists the natural and essential role of a referee/coodinator, namely the wireless service provider with its infrastructure. First, direct interactions between users and the base station can well carry the iterative steps of Gur Game. As shown in Fig. 1, the network platform guides each user to reach the desired state gradually. Second, in a real-world deployment, technologies in mobile multi-hop relay networks can also be applied, e.g., multi-hop relaying with IEEE 802.16j [32] can improve coverage and capacity issues of the crowdsourcing system.

B. Fully Charged Device Battery

In Section IV-A, when modeling the case study from the data set, we assume that user devices are fully charged at the beginning of any of the total 91 independent time periods. This assumption is only for the ease of calculating the phone remaining energy level, since its battery status information is not provided in the data set. With this assumption and the regular phone usage such voice call, SMS, and standby, which are computable from the data set, we can deduce their energy state when receiving the crowdsourcing request. It is worth noting that given the user's initial energy state, our goal is to balance their consumed energy during crowdsourcing process. Absolutely, we can set random energy value for each user device at the beginning of each time periods, and it has no impact on our proposed optimization strategy.

C. Truthfulness of the Bidding Process

In Section VI, reverse auction is used as the main framework of our incentive-base participant selection mechanism. In this paper, our goal is to maximize the benefit of the network platform when offering fair returns to participants. Based on bidding information, we focus on how to select proper participants to minimize the total paid credits and achieve the required QoI levels. However, truthfulness is a critical property of any auction scheme. In truthful bidding, no buyer can improve its utility by submitting a bid different from its true valuation, no matter how others submit [33]. If this property is not guaranteed, the auction could be vulnerable to malicious manipulation and produce very poor outcomes. For the truthful bidding implementation, well-known truthful auction schemes like Vickrey-Clarke-Groves (VCG) scheme [34], [35], [36] and McAfee double auction [37] can be integrated with our existing scheme. Take VCG auction for example. For a set of auctioned item $\mathcal{M} = \{t_1, t_2, ..., t_m\}$ and a set of bidders $\mathcal{N} = \{b_1, b_2, ..., b_n\}$, let $V_{\mathcal{N}}^{\mathcal{M}}$ be the social value of auction for a given bidding combination. In VCG auction, the bidder b_i who wins the item t_j needs to pay the social cost of his winning that is incurred by the rest of the bidders, namely $V_{\mathcal{N}\setminus\{b_i\}}^{\mathcal{M}} - V_{\mathcal{N}\setminus\{b_i\}}^{\mathcal{M}\setminus\{t_j\}}$. It is proved that under this scheme, to achieve the maximization of net utility, a bidder should use his true valuations for the auctioned items. This method can be merge into our existing participant selection scheme, since given the bidding information of all users, by Algorithm 2, the credit processing unit of network platform can check the impact of each user's participation on the welfare of rest users, and hence calculate the social cost of each user's winning.

IX. Conclusion

In this paper, we propose a novel QoI-aware, energy-efficient participatory crowdsourcing framework, powered by the distributed decision-making process of Gur Game. Our solution fully considers the QoI requirements of the request while providing a satisfactory level of total energy consumption fairness among all participants, and most importantly, in a distributed manner. Specifically, we largely extend the traditional framework of Gur Game by merging multiple automaton chains into a single chain with multiple steady states, representing different amount of information contribution per user. We propose a two-step decision making algorithm to meet the requirements of both QoI (Q-step) and energy fairness (V-step). We also propose an incentive-based participant selection scheme to maximize the platform's benefits and provide satisfactory credits to the participants. Extensive experimental results on the MIT Social Evolution data set show that the proposed scheme successfully fulfills the QoI requirements of the request, while providing a satisfactory level of energy consumption fairness among participants and achieving the credit saving, with negligible computational complexity.

References

[1] K. K. Rachuri, M. Musolesi, C. Mascolo, P. J. Rentfrow, C. Longworth, and A. Aucinas, "Emotionsense: a mobile phones based adaptive platform for experimental social psychology research," in *ACM Ubicomp'10*, 2010, pp. 281–290.

[2] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava, "Participatory sensing," in *Workshop on World-Sensor-Web (WSW'06): Mobile Device Centric Sensor Networks and Applications*, 2006, pp. 117–134.

[3] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn, "The rise of people-centric sensing," *IEEE Internet Computing*, vol. 12, no. 4, pp. 12–21, 2008.

[4] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *IEEE Comm. Mag.*, vol. 48, no. 9, pp. 140–150, Sept. 2010.

[5] J. Howe, "The rise of crowdsourcing," *Wired*, vol. 14, no. 6, pp. 1–4, 2006.

[6] D. C. Brabham, "Crowdsourcing as a model for problem solving an introduction and cases," *Convergence: the international journal of research into new media technologies*, vol. 14, no. 1, pp. 75–90, 2008.

[7] R. Bonney, C. B. Cooper, J. Dickinson, S. Kelling, T. Phillips, K. V. Rosenberg, and J. Shirk, "Citizen science: A developing tool for expanding science knowledge and scientific literacy," *BioScience*, vol. 59, no. 11, pp. 977–984, 2009.

[8] J. M. DiMicco, "Mobile ad hoc voting," in *ACM CHI'02 Workshop on Mobile Ad-Hoc Collaboration*, April 2002.

[9] M. E. Johnson and K. C. Chang, "Quality of information for data fusion in net centric publish and subscribe architectures," in *FUSION'05*, July 2005, pp. 1070–1077.

[10] C. Bisdikian, L. M. Kaplan, M. B. Srivastava, D. J. Thornley, D. Verma, and R. I. Young, "Building principles for a quality of information specification for sensor information," in *FUSION'09*, July 2009, pp. 1370–1377.

[11] C. H. Liu, P. Hui, J. W. Branch, C. Bisdikian, and B. Yang, "Efficient network management for context-aware participatory sensing," in *IEEE SECON'11*, June 2011, pp. 116–124.

[12] J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson, "Easytracker: automatic transit tracking, mapping, and arrival time prediction using smartphones," in *ACM SenSys'11*, 2011, pp. 68–81.

[13] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell, "Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application," in *ACM SenSys'08*, 2008, pp. 337–350.

[14] S. Gaonkar, J. Li, R. R. Choudhury, L. Cox, and A. Schmidt, "Micro-blog: sharing and querying content through mobile phones and social participation," in *ACM MobiSys'08*, 2008, pp. 174–186.

[15] N. D. Lane, D. Lymberopoulos, F. Zhao, and A. T. Campbell, "Hapori: context-based local search for mobile phones using community behavioral modeling and similarity," in *ACM Ubicomp'10*, 2010, pp. 109–118.

[16] Y. Lee, Y. Ju, C. Min, S. Kang, I. Hwang, and J. Song, "Comon: cooperative ambience monitoring platform with continuity and benefit awareness," in *ACM MobiSys'12*, 2012, pp. 43–56.

[17] L. Jordan, A. Stallins, S. Stokes IV, E. Johnson, and R. Gragg, "Citizen mapping and environmental justice: Internet applications for research and advocacy," *Environmental Justice*, vol. 4, no. 3, pp. 155–162, 2011.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2367029, IEEE Transactions on Vehicular Technology

[18] N. Maisonneuve, M. Stevens, M. Niessen, P. Hanappe, and L. Steels, "Citizen noise pollution monitoring," in *the 10th Annual Int'l Conf. on Digital Government Res.: Social Netw.: Making Conn. between Citizens, Data and Government*, 2009, pp. 96–103.

[19] R. Iyer and L. Kleinrock, "QoS control for sensor networks," in *IEEE ICC'03*, June 2003, pp. 517–521.

[20] M. Tsetlin, "Finite automata and modeling the simplest forms of behavior," Ph.D. dissertation, V.A. Steklov Mathematical Institute, 1964.

[21] B. Tung and L. Kleinrock, "Distributed control methods," in *2nd Int'l Sym. on High Perf. Dist. Comp.*, Jul 1993, pp. 206–215.

[22] ——, "Using finite state automata to produce self-optimization and self-control," *IEEE Transactions on Parallel and Distributed Systems*, vol. 7, no. 4, pp. 439–448, 1996.

[23] L. Zhao, C. Xu, Y. Xu, and X. Li, "Energy-aware QoS control for wireless sensor network," in *1st IEEE Conf. on Industrial Electronics and App.*, May 2006, pp. 1–6.

[24] S. I. Nayer and H. H. Ali, "A dynamic energy-aware algorithm for self-optimizing wireless sensor networks," *Springer LNCS: Self-Organizing Sys.*, vol. 5343, pp. 262–268, 2008.

[25] J.-S. Lee and B. Hoh, "Dynamic pricing incentive for participatory sensing," *Pervasive Mob. Comput.*, vol. 6, no. 6, pp. 693–708, Dec. 2010.

[26] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing," in *ACM Mobicom'12*, 2012, pp. 173–184.

[27] T. Luo and C.-K. Tham, "Fairness and social welfare in incentivizing participatory sensing," in *IEEE SECON'12*, 2012, pp. 425–433.

[28] J. J. Clark and A. L. Yuille, *Data Fusion for Sensory Information Processing Systems*. Kluwer Academic Publishers, 1990.

[29] A. Madan, M. Cebrian, S. Moturu, K. Farrahi, and A. Pentland, "Sensing the 'Health State' of a community," *IEEE Perv. Comp.*, vol. 11, no. 4, pp. 36–45, Oct. 2012.

[30] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2003.

[31] J. Lofberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *the CACSD Conf.*, 2004, pp. 284–289. [Online]. Available: http://users.isy.liu.se/johanl/yalmip

[32] V. Genc, S. Murphy, Y. Yu, and J. Murphy, "IEEE 802.16J relay-based wireless access networks: an overview," *IEEE Wireless Communications*, vol. 15, no. 5, pp. 56–63, October 2008.

[33] P. Klemperer, "What really matters in auction design," *The Journal of Economic Perspectives*, vol. 16, no. 1, pp. 169–189, 2002.

[34] E. H. Clarke, "Multipart pricing of public goods," *Public choice*, vol. 11, no. 1, pp. 17–33, 1971.

[35] T. Groves, "Incentives in teams," *Econometrica: Journal of the Econometric Society*, pp. 617–631, 1973.

[36] W. Vickrey, "Counterspeculation, auctions, and competitive sealed tenders," *The Journal of finance*, vol. 16, no. 1, pp. 8–37, 1961.

[37] R. P. McAfee, "A dominant strategy double auction," *Journal of economic Theory*, vol. 56, no. 2, pp. 434–450, 1992.

Chi Harold Liu (M'10) is a Full Professor at the School of Software, Beijing Institute of Technology, China. He holds a Ph.D. degree from Imperial College, UK, and a B.Eng. degree from Tsinghua University, China. Before moving to academia, he worked several years with IBM Research - China, Deutsche Telekom Laboratories, and IBM T. J. Watson Research Center, USA. His current research interests include the Internet-of-Things (IoT), Big Data analytics, and Cloud Computing. He received the Distinguished Young Scholar Award in 2013, IBM First Plateau Invention Achievement Award in 2012, and IBM First Patent Application Award in 2011 and was interviewed by EEWeb.com as the Featured Engineer in 2011. He has published more than 60 prestigious conference and journal papers and owned more than 10 EU/U.S./China patents. He serves as the editor for KSII Trans. on Internet and Information Systems and the book editor for six books published by Taylor & Francis Group, USA. He also has served as the general chair of three IoT workshop with IEEE SECON'13/IEEE WCNC'12/ACM UbiComp'11. He served as the consultant to Asian Development Bank, Bain & Company, and KPMG, USA, and the peer reviewer for Qatar National Research Foundation, and National Science Foundation, China. He is a member of IEEE and ACM.

Jun Fan receives his MSc degree from Department of Computer Science at Beijing Institute of Technology, China. He is currently a Software Engineer with Baidu Corp. Ltd., and his research interests include cooperative communication, vehicular ad hoc networks and the Internet-of-Things (IoT).

Pan Hui (M'07) received his Ph.D degree from Computer Laboratory, University of Cambridge, and his MPhil and BEng both from the EEE Department at University of Hong Kong. He is currently a faculty member of the CSE Department at the HKUST where he directs the System and Media Lab. He also serves as a Distinguished Scientist of Telekom Innovation Laboratories (T-labs) Germany and an adjunct Professor of social computing and networking at Aalto University, Finland. He also has spent several years in T-labs and Intel Research Cambridge. He has published more than 100 research papers and has several granted and pending EU patents. He has founded and chaired several IEEE/ACM conferences/workshops, and served on the technical program committee of numerous international conferences and workshops including IEEE Infocom, SECON, MASS, Globecom, WCNC, and ITC.

Jie Wu (F'09) is the chair and a Laura H. Carnell Professor in the Department of Computer and Information Sciences at Temple University. Prior to joining Temple University, he was a program director at the National Science Foundation and Distinguished Professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. Dr. Wu regularly published in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE Transactions on Computers, IEEE Transactions on Service Computing, and Journal of Parallel and Distributed Computing. Dr. Wu was general co-chair/chair for IEEE MASS 2006 and IEEE IPDPS 2008 and program co-chair for IEEE INFOCOM 2011. Currently, he is serving as general chair for IEEE ICDCS 2013 and ACM MobiHoc 2014, and program chair for CCF CNCC 2013. He was an IEEE Computer Society Distinguished Visitor, ACM Distinguished Speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a CCF Distinguished Speaker and a Fellow of the IEEE. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award.

Kin K. Leung (F'01) received the BS degree from the Chinese University of Hong Kong in 1980, and the MS and PhD degrees in computer science from the University of California, Los Angeles, in 1982 and 1985, respectively. He started his career at AT&T Bell Labs, then AT&T Labs, and Bell Labs of Lucent Technologies. Since 2004, he has been the Tanaka Chair Professor of Internet technology at Imperial College. His research interests include radio resource allocation, MAC protocol, TCP/IP protocol, mobility management, network architecture, realtime applications, and teletraffic issues for broadband wireless networks. He received the Distinguished Member of Technical Staff Award from AT&T Bell Labs in 1994, and was a co-recipient of the 1997 Lanchester Prize Honorable Mention Award. He holds the Royal Society Wolfson Research Merit Award from 2004 to 2009. He is a Fellow of the IEEE.